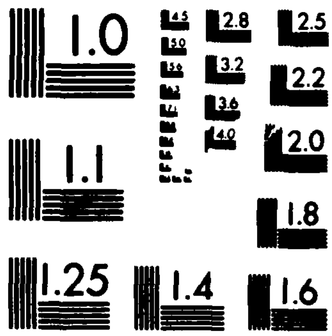
AD-A179 952 EVIDENCE COMBINATION USING LIKELIHOOD GENERATORS(U) 1/1
ROCHESTER UNIV NY DEPT OF COMPUTER SCIENCE D SHER
JAN 87 TR-192 DACA76-85-C-0001

UNCLASSIFIED F/G 12/9 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Evidence Combination Using Likelihood Generators

David Sher
Computer Science Department
The University of Rochester
Rochester, New York 14627

January 1987
TR192

Department of Computer Science
University of Rochester
Rochester, New York 14627

Evidence Combination Using Likelihood Generators

David Sher
Computer Science Department
The University of Rochester
Rochester, New York 14627

January 1987
TR192

Abstract

The author

~~Here, I~~ address the problem of combining output of several detectors for the same feature of an image. ~~I~~ shows that if the detectors return likelihoods I can robustly combine their outputs. The combination has the advantages that:

- The confidences of the operators in their own reports are taken into account. Hence if an operator is confident about the situation and the others are not then the reports of the confident operator dominates the decision process;
- *A priori* confidences in the different operators can be taken into account;
- The work to combine 'N' operators is linear in 'N'.

This theory has been applied to the problem of boundary detection. Results from these tests are presented here.

This work would have been impossible without the advice and argumentation of such people as Paul Chou and Mike Swain (who has made suggestions from the beginning) and of course my advisor Chris Brown. and who could forget Jerry Feldman. This work was supported by the Defense Advanced Research Projects Agency U. S. Army Engineering Topographic Labs under grant number DACA76-85-C-0001. Also this work was supported by the Air Force Systems Command. Rome Air Development Center. Griffiss Air Force Base. New York 13441-5700. and the Air Force Office of Scientific Research. Bolling AFB. DC 20332, under Contract No. F30602-85-C-0008. This contract supports the Northeast Artificial Intelligence Consortium (NAIC).

A179952

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER TR 192	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Evidence Combination Using Likelihood Generators		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) David Sher		8. CONTRACT OR GRANT NUMBER(s) DACA76-85-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computer Science Department The University of Rochester Rochester, New York 14627		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS DARPA/1400 Wilson Blvd Arlington, VA 22209		12. REPORT DATE January 1987
		13. NUMBER OF PAGES 29
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Office of Naval Research Information Systems Arlington, VA 22217		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this document is unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

evidence combination, Bayesian reasoning, likelihood

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Here, I address the problem of combining output of several detectors for the same feature of an image. I show that if the detectors return likelihoods I can robustly combine their outputs. The combination has the advantages that:

* The confidences of the operators in their own reports are taken into account. Hence if an operator is confident about the situation and

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued)

the others are not then the reports of the confident operator dominates the decision process.

* A priori confidences in the different operators can be taken into account

* The work to combine 'N' operators is linear in 'N'.

This theory has been applied to the problem of boundary detection. Results from these tests are presented here.

1. Introduction

Often in computer vision one has a task to do such as deriving the boundaries of objects in an image or deriving the surface orientation of objects in an image. Often one also has a variety of techniques to do this task. For boundary detection there are a variety of techniques from classical edge detection literature [Ballard82] and the image segmentation literature e.g. [Ohlander79]. For determining surface orientation there are techniques that derive surface orientation from intensities [Horn70] and texture [Ikeuchi80] [Aloimonos85]. These techniques make certain assumptions about the structure of the scene that produced the data. Such techniques are only reliable when their assumptions are met. Here I show that if several algorithms return likelihoods I can derive from them the correct likelihood when at least one of the algorithms' assumptions are met. Thus I derive an algorithm that works well when any of the individual algorithms works well.

The mathematics here were derived independently but are similar to the treatment in [Good50]. and [Good83], using different notation. To understand my results first one must understand the meaning of likelihood.

2. Likelihoods

In this paper I call the assumptions that an algorithm makes about the world a *model*. Most models for computer vision problems describe how configurations in the real world generate observed data. Because imaging projects away information, the models do not explicitly state how to derive the configuration of the real world from the sensor data. As a result, graphics problems are considerably easier than vision problems. Programs can generate realistic images that no program can analyze.

Let O be the observed data, f a feature of the scene whose existence we are trying to determine (like a boundary between two pixels) and M a model. Many computer vision problems can be reduced to finding the probability of the feature given the model and the data, $P(f|O\&M)$. However most models for computer vision instead make it easy to compute $P(O|f\&M)$. I call $P(O|f\&M)$ (inspired by the statistical literature) the *likelihood* of f given observed data O under M. As an example assume f is "the image has a constant intensity before noise". M says that the image has a normally distributed uncorrelated (between pixels) number added to each pixel (the noise). Calculating $P(O|M\&f)$ is straight-forward (a function of the mean and variance of O).

A theorem of probability theory, *Bayes' law*, shows how to derive conditional probabilities for features from likelihoods and prior probabilities. Bayes' law is shown in equation 1.

$$P(f|O\&M)=\frac{P(O|f\&M)P(f|M)}{P(O|f\&M)P(f|M)+P(O|\sim f\&M)P(\sim f|M)} \tag{1}$$

f is the feature for which we have likelihoods. M is the domain model we are using. $P(O|f\&M)$ is the likelihood of f under M and $P(f|M)$ is the probability under M of f

For features that can take on several discrete mutually exclusive labels (rather than just true and false) such as surface orientation (which can be a pair of angles to the nearest degree or "not applicable" (at boundaries)) a more complex form of Bayes' law shown in equation 2 yields conditional probabilities from likelihoods and priors.

$$P(l|O\&M) = \frac{P(O|l\&M)P(l|M)}{\sum_{l' \in L(f)} P(O|l'\&M)P(l'|M)} \tag{2}$$

l is a label for feature f and $L(f)$ is the set of all possible labels for feature f.

Another important use for explicit likelihoods is for use in Markov random fields. Markov random fields describe complex priors that can capture important information. Several people have applied Markov random fields to vision problems [Geman84]. Likelihoods can be used in a Markov random field formulation to derive estimates of boundary positions [Marroquin85b] [Chou87]. In [Sher86] and [Sher87] I discuss algorithms for determining likelihoods of boundaries.

Let us call an algorithm that generates likelihoods a *likelihood generator*. Different models lead to different likelihood generators. The difference between two likelihood generators' models can be a single constant (such as the assumed standard deviation of the noise) or the two likelihood generators' models may not resemble each other in the slightest.

Consider likelihood generators L_1 and L_2 with models M_1 and M_2 and assume they both determine probability distributions for the same feature. L_1 can be considered to return the likelihood of a label l for feature f given observed data O and the domain model M_1. Thus L_1 calculates $P(O|f=l\&M_1)$. Also L_2 calculates $P(O|f=l\&M_2)$. A useful combination of L_1 and L_2 is the likelihood detector that returns the likelihoods for the case where M_1 or M_2 is true. Also the prior confidences one has in M_1 and M_2 should be taken into account.

This paper studies deriving $P(O|f=1\&(M_1 \vee M_2))$. Note that if I can derive rules for combining likelihoods for two different models then by applying the combination rules N times, N likelihoods are combined. Thus all that is needed is combination rules for two models.

3. Combining Likelihoods From Different Models

To combine likelihoods derived under M_1 and M_2 an examination of the structure and interaction of the two models is necessary. M_1 and M_2 must have the same definition for the feature being detected. If the feature is defined differently for M_1 and M_2 then M_1 and M_2 are about different events, and the likelihoods can not be combined with the techniques developed in this section.

Thus the likelihood generated by an occlusion boundary detector can not be combined with the likelihood generated by a detector for boundaries within the image of an object (such as corners internal to the image). A detector of the likelihood of heads on a coin flip can not be combined with a detector of the likelihood of rain outside using this theory. (However easy it may be using standard probability theory.)

If the labeling of a feature f implies a labeling for another feature g then in theory one can combine a f detector with a g detector by using the g detector that is implied by the f detector. As an example a region grower could be combined with a boundary detector since the position of the regions implies the positions of the boundaries.

3.1. Combining Two Likelihoods

The formula for combining the likelihoods generated under M_1 and M_2 requires prior knowledge. Necessary are the prior probabilities $P(M_1)$ and $P(M_2)$ that the domain models M_1 and M_2 are correct as well as $P(M_1\&M_2)$. Often $P(M_1\&M_2) = 0$. When this occurs the two models contradict each other. I call two such models *disjoint* because both can not describe the situation

simultaneously. If M_1 is a model with noise of standard deviation $4 \pm \varepsilon$ and M_2 is a model with noise of standard deviation $8 \pm \varepsilon$ then their assumptions contradict and $P(M_1 \& M_2) = 0$.

Prior probabilities for the feature labels under each model ($P(f=l|M_1)$ and $P(f=l|M_2)$) are necessary. If $P(M_1 \& M_2) \neq 0$ then the prior probability of the feature label under the conjunction of M_1 and M_2 ($P(f=l|M_1 \& M_2)$) and the output of a likelihood generator for the conjunction of the two models ($P(O|f=l \& (M_1 \& M_2))$) are needed. If I have this prior information I can derive $P(O|f=l \& (M_1 \vee M_2))$.

If I were to combine another model, M_3, with this combination I need the priors $P(M_3)$, $P(f|M_3)$, $P(M_3 \& (M_1 \vee M_2)$ and $P(f|M_3 \& (M_1 \vee M_2))$. To add on another model I need another 4 priors. Thus the number of prior probabilities to combine n models is linear in n.

Thus all that is left is to derive the combination rule for likelihood generators given this prior information. The derivation starts by applying the definition of conditional probability in equation 3.

$$P(O|f=l \& (M_1 \vee M_2)) = \frac{P(O \& f=l \& (M_1 \vee M_2))}{P(f=l \& (M_1 \vee M_2))} \tag{3}$$

The formula for probability of a disjunction is applied to the numerator and denominator in equation 4.

$$P(O|f=l \& (M_1 \vee M_2)) = \frac{P(O \& f=l \& M_1) + P(O \& f=l \& M_2) - P(O \& f=l \& M_1 \& M_2)}{P(f=l \& M_1) + P(f=l \& M_2) - P(f=l \& M_1 \& M_2)} \tag{4}$$

In equation 5 the definition of conditional probability is applied again to the terms of the numerator and the denominator.

$$P(O|f=l \& (M_1 \vee M_2)) = \frac{\left|\begin{array}{c} P(O|f=l \& M_1)P(f=l|M_1)P(M_1) \\ + \\ P(O|f=l \& M_2)P(f=l|M_2)P(M_2) \\ - \\ P(O|f=l \& M_1 \& M_2)P(f=l|M_1 \& M_1)P(M_1 \& M_2) \end{array}\right|}{P(f=l|M_1)P(M_1) + P(f=l|M_2)P(M_2) - P(f=l|M_1 \& M_2)P(M_1 \& M_2)} \tag{5}$$

Different assumptions allow different simplifications to be applied to the rule in equation 5. If the two models are disjoint equation 5 reduces to equation 6.

$$P(O|f=l \& (M_1 \vee M_2)) = \frac{\left|\begin{array}{c} P(O|f=l \& M_1)P(f=l|M_1)P(M_1) \\ + \\ P(O|f=l \& M_2)P(f=l|M_2)P(M_2) \end{array}\right|}{P(f=l|M_1)P(M_1) + P(f=l|M_2)P(M_2)} \tag{6}$$

Another assumption that simplifies things considerably is the assumption that prior probabilities for all feature labelings in all the models and combinations thereof are the same. I call this assumption *constancy of priors*. When constancy of priors is assumed $P(f=l|M_1) = P(f=l|M_2) = P(f=l|M_1 \& M_2)$. Making this assumption reduces the number of priors that need to be determined. Since determining prior probabilities from a model is sometimes a difficult task the constancy of priors is a useful simplification. With constancy of priors equation 5 reduces to equation 7.

$$P(O|f=l\&(M_1 \vee M_2)) = \frac{\begin{bmatrix} P(O|f=l\&M_1)P(M_1) \\ + \\ P(O|f=l\&M_2)P(M_2) \\ - \\ P(O|f=l\&M_1\&M_2)P(M_1\&M_2) \end{bmatrix}}{P(M_1)+P(M_2)-P(M_1\&M_2)} \tag{7}$$

Equation 6 with constancy of priors reduces to equation 8.

$$P(O|f=l\&(M_1 \vee M_2)) = \frac{\begin{bmatrix} P(O|f=l\&M_1)P(M_1) \\ + \\ P(O|f=l\&M_2)P(M_2) \end{bmatrix}}{P(M_1)+P(M_2)} \tag{8}$$

Thus equation 8 describes the likelihood combination rule with disjoint models and constancy of priors.

3.2. Understanding the Likelihood Combination Rule

The easiest incarnation of the likelihood combination rule to understand is the rule for combining likelihoods from disjoint models given constancy of priors across models (equation 8). Here the combined likelihood is the weighted average of the likelihoods from the individual models weighted by the probabilities of the models applying. (The combined likelihood is the likelihood given the disjunction of the models).

If models M_1 and M_2 are considered equally probable and the likelihoods returned by M_1's detector are considerably larger than those of M_2's detector then the probabilities determined from the combination of M_1 and M_2 are close to those determined from M_1. Thus a model with large likelihoods determines the probabilities. To illustrate this principle consider an example.

Assume that a coin has been flipped $n+1$ times. The results of flipping it has been reported for the first n times. The task is to determine the probability of heads having been the result of the $n+1^{st}$ flip. Consider the results of each coin flip independent. Let M_1 be the coin being fair so that the probability of heads and tails is equal. Let M_2 be that the coin is biased with the probability of heads is π and tails $1-\pi$ with π being a random choice with equal probability between p and $1-p$. Hence the coin is biased towards heads or tails with equal probability but the bias is consistent between coin tosses. The probability of heads remains the same for all coin tosses in both models. M_1 and M_2 are disjoint (the coin is either fair or it isn't but not both) and the prior probability of a flip being heads or tail is the same for both, .5.

Under M_1 the probability of each of the possible flips of $n+1$ coins is 2^{-n-1}. Under M_2 the probability of $n+1$ flips of coins with h heads and $t=n+1-h$ tails is:

$$\tfrac{1}{2}p^h(1-p)^t+\tfrac{1}{2}p^t(1-p)^h$$

Let $n=2$ and $p=.9$. Assume the first two flips are both heads. Let H be "the third flip was heads" and T be "the third flip was tails." The likelihood of H given the observed data is the probability of all 3 flips being heads divided by the probability of the third flip being heads. The likelihood of T given the observed data is the probability of the first 2 being heads and the 3rd tails divided by the probability of the third flip being tails.

Under M_1 the probability of all 3 flips being heads is 0.125 and the probability of a flip being heads is 0.5 thus the likelihood of H is 0.25. The likelihood of T is 0.25 by the same reasoning.

Applying Bayes' law to get the probability of H under M_1 one derives a probability of .5 .

Under M_2 the probability of all 3 flips being heads is 0.365 and the probability of a flip being heads is 0.5. Thus the likelihood of H is 0.73. Under M_2 the probability of the first two being heads and the third being tails is 0.045 and the probability of a flip being tails is 0.5. Thus the likelihood of T is 0.09. Applying Bayes' law under M_2 a probability of H being 0.89 is derived.

If M_1 and M_2 are considered equally probable then the combination of the likelihoods from the two models is the average of the two likelihoods. Thus the likelihood of H for this combination is 0.49 and the likelihood of T is 0.17 (likelihoods don't have to sum to 1). Bayes' law combines these probabilities to get 0.74 for the 3[rd] flip to be heads.

The table in figure 1 describes combining various M_2's with different values of p with M_1 for the different combinations with $n=4$

Observed Coin Flips	Combined with M_1 or just M_2	Likelihood of H $p=.6$	Likelihood of H $p=.9$	Likelihood of T $p=.6$	Likelihood of T $p=.9$	Probability of H $p=.6$	Probability of H $p=.9$
HHHH	Just M_2	0.088	0.5905	0.0672	0.0657	0.567	0.8999
	Combined	0.07525	0.3265	0.06485	0.0641	0.537	0.8359
HHHT	Just M_2	0.0672	0.0657	0.0576	0.0081	0.5385	0.8902
	Combined	0.06485	0.0641	0.06005	0.0353	0.5192	0.6449
HHTT	Just M_2	0.0576	0.0081	0.0575	0.0081	0.5	0.5
	Combined	0.06005	0.0353	0.06	0.0353	0.5	0.5
HTTT	Just M_2	0.0576	0.0081	0.0672	0.0657	0.4615	0.1098
	Combined	0.06005	0.0353	0.06485	0.0641	0.4808	0.3551
TTTT	Just M_2	0.0672	0.0657	0.088	0.5905	0.433	0.1001
	Combined	0.06485	0.0641	0.07525	0.3265	0.4629	0.1641

Figure 1: Result of likelihood combination Rule

Look at the probabilities with $p=.9$ and the observed data is HHHH. For this case the observed data fits M_2 much better than M_1 and the probability from combining M_1 and M_2 is close to the probability resulting from using just M_2, .9. If we had a longer run of heads the probability of future heads would approach exactly M_2's prediction, .9. On the other hand if we had a long run of equal numbers of heads and tails the probability of future heads would quickly approach the prediction of M_1, .5. When the observed data is HHHT the observed data fits M_1 about as well as M_2 and the resulting probability is near the average of .5 predicted by M_1 and 0.8902 predicted by M_2. Thus when the observed data is a good fit for a particular model (like M_2) the probabilities predicted by the combination is close to the probabilities predicted by the fitted model. If two models fit about equally then the result is an average of the probabilities[1].

4. When No Model Applies

Given a set of likelihood generators and their models, using the evidence combination described in section 3 we can get the likelihood for the feature labelings given that at least one model applies. Thus if we have likelihoods of a boundary given models with the noise standard

[1]However the feature that the decision theory predicts is not the average of the features predicted under the two different models in general.

deviations near to 4, 8 and 16 in them we can derive the likelihood of a given the noise standard deviation is near to 4 or 8 or 16 (no matter which). Thus we can derive the probability distribution over feature labelings given that at least one of our models applies. However what we are trying to derive is the *physical probability distribution* over the feature labelings. This is the probability distribution over feature labels given the observed data (estimated by the long run frequencies over the feature labels given the observed data). The problem is that there may be a case where none of the models assumptions is true. In the Venn diagram of figure 2 each set represents the set of situations where a model's assumptions are true. The area marked NO MODEL is the set of situations where all the models fail.

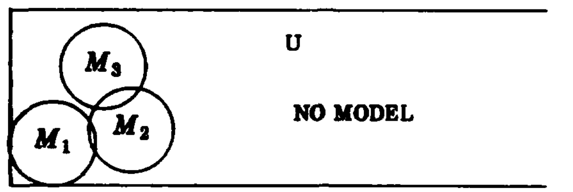


Figure 2: Venn Diagram of Models

What should the likelihood of a feature label be if no model applies? To answer this question I examine the companion question of what should the probability of a feature label be if no model applies. Assume a prior probability for the label is available. If a posterior probability is different from a prior probability for the feature then information has been added to get the posterior. (Only information can justify changing from the prior.) Since having no model means intuitively having no information then the posterior should be the same as the prior. If and only if the likelihoods of all feature labels are equal, the posterior probability is the same as the prior. Hence the likelihoods of the feature labels should be equal for any particular piece of observed data. In this section I assume a prior proability distribution is a available over feature labels. If no such distribution is available an uninformative prior can be constructed [Frieden85].

To constrain the problem further, consider whether any piece of observed data should be more probable than any other when no model applies. It seems unreasonable that one could conclude that some observations are more probable than others without any model of how those observations were produced. Hence all the likelihoods should be equal. This constraint is sufficient to determine the likelihoods when no model applies. I think that this solution minimizes cross entropy with the prior (since it returns the prior) [Johnson85].

To derive the physical probability distribution over feature labels, the "no model" likelihoods should be combined with the likelihoods derived for the models. The probability of each of the models and their combinations must have been available to use the combination rules from section 3. Hence the probability that one or more of the models applies is known. The probability of no model is 1 minus that probability. The conjunction of some model applying and no model applying has 0 probability. Hence combination rule 6 can be applied to derive the likelihoods under any conditions from the likelihoods for any model applying.

As example consider the problem of seeing HHHH and trying to derive the probability of a fifth head given the equally likely choices that the coin is fair or is biased to .9 (biased either for heads or tails with equal probability). The combined likelihood of H is 0.3265 (from figure 1). The combined likelihood of T is 0.0641. As an example, assume that the probabilities that the assumptions of M_1 were true was 0.4 and similar for M_2. Then 0.4 of the time we feel the coin is

fair, 0.4 of the time we feel it has been biased by 0.9, and 0.2 of the time we have no model about what happened. The likelihood of HHHH under "NO MODEL" is .0625 regardless of H or T (Since the likelihood of all 4 coin flip events are equal and must sum to 1). Combining the "NO MODEL" likelihoods with likelihoods of 0.2737 for H and 0.06378 for T (see figure 1), the probability of H from applying Bayes' law to these likelihoods is 0.811. This probability is somewhat nearer to .5 than the probability of 0.8359 derived without taking the possibility of all the models failing into account.

Taking the possibility of all models failing lends certain good properties to the system. Probabilities of 0 or 1 become impossible without priors of 0 or 1. Thus the system is denied total certainty. Numbers near 0 or 1 cause singularities in the equations under finite precision arithmetic. Total certainty represents a willingness to ignore all further evidence. I find that property undesirable in a system. Denying the system total certainty also results in the property that the system must have all probability distribution over feature labels between ε and $1-\varepsilon$ for an ε proportional to the probability that no model applies. Thus there is a limit to how certain our system is about any feature labeling in our uncertain world.

5. Results

I have applied this evidence combination to the boundary detection likelihood generators described in [Sher87]. Here I prove my claims that the evidence combination theory allows me to take a set of algorithms that are effective but not robust and derive an algorithm that is robust. The output of such an algorithm is almost as good as the best of its constituents (the algorithms that are combined).

5.1. Artificial Images

Artificial images were used to test the algorithms described in section 3 quantitatively. I used as a source of likelihoods the routines described in [Sher87]. Because the positions of the boundaries in an artificial image are known one can accurately measure false positive and negative rates for different operators. Also one can construct artificial images to precise specifications. The artificial images I use is an image composed of overlapping circles with constant intensity and aliasing at the boundaries shown in figure 3.

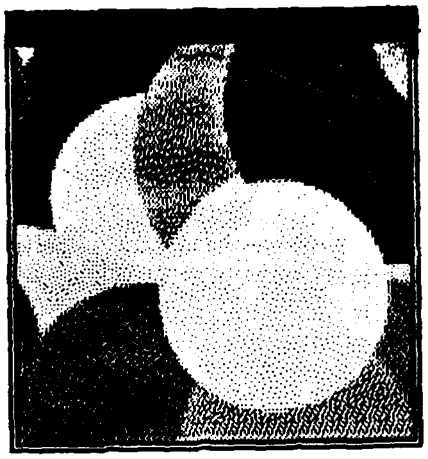
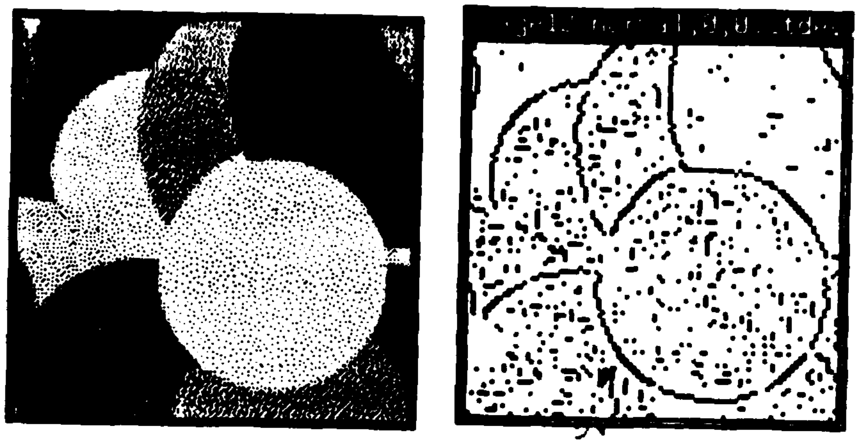


Figure 3: Artificial Test Image

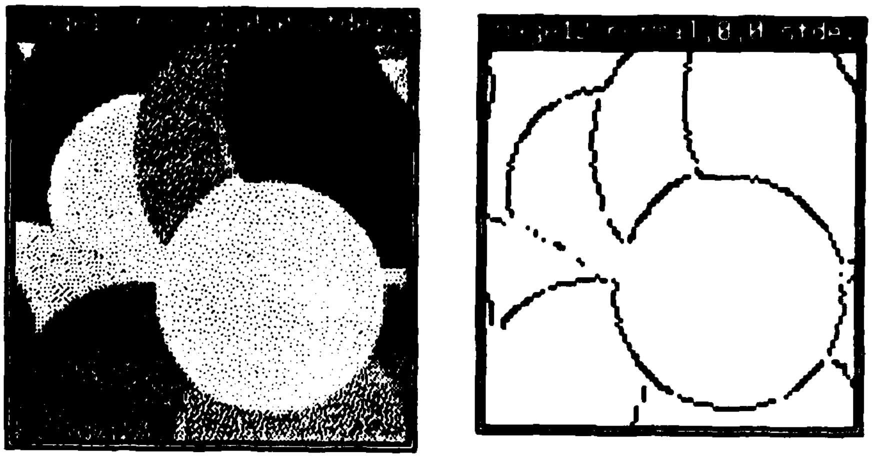
The intensities of the circles were selected from a uniform distribution from 0 to 254. To the circles were added normally distributed uncorrelated noise with standard deviations 4, 8, 12, 16, 20, and 32. The software to generate images of this form was built by Myra Van Inwegen working under my direction. This software will be described in an upcoming technical report.

In figure 4 I show the result of applying the detector tuned to standard deviation 4 noise to the artificial image with standard deviation 12 noise added to it. In figure 5 I show the result of applying the detector tuned to standard deviation 12 noise to an image with standard deviation 12 noise added to it. In figure 6 I show the result of applying the combination of the detectors tuned to 4, 8, 12, and 16 standard deviation noise. The combination rule was that for disjoint models with the *same priors*. The 4 models were combined with equal probability. These operator outputs are thresholded at 0.5 probability with black indicating an edge and white indicating no edge.



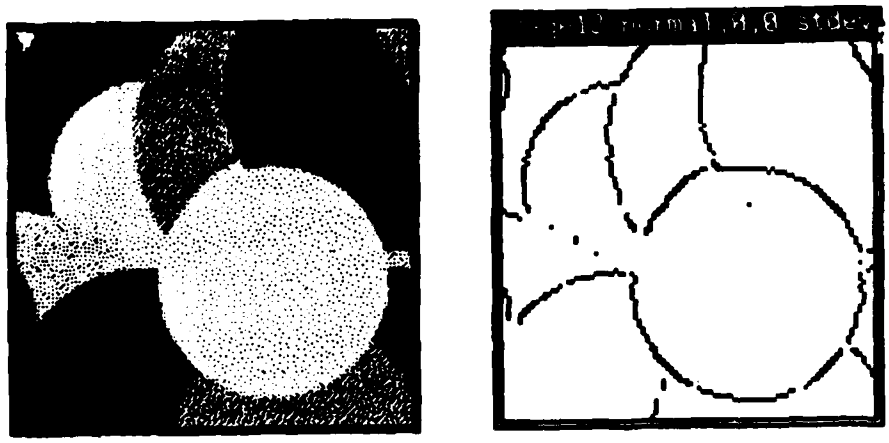
a: Image with $\sigma=12$ noise b: Output of $\sigma=4$ detector

Figure 4: $\sigma=4$ detector applied to 3 image with $\sigma=12$ noise



a: Image with $\sigma=12$ noise b: Output of $\sigma=12$ detector

Figure 5: $\sigma=12$ detector applied to 3 image with $\sigma=12$ noise



a: Image with $\sigma=12$ noise b: Output of combined detector
Figure 6: Combined detector applied to 3 image with $\sigma=12$ noise

Note that the result of using the combined operator is similar to that of the operator tuned to the correct noise level. Most of the false boundaries found by the $\sigma=4$ operator are ignored by the combined operator.

Using this artificial image I have acquired statistics about the behavior of the combined detector vs the tuned ones under varying levels of noise. Figure 7 shows the false positive rate for the detector tuned to standard deviation 4 noise as the noise in the image increases[2]. Figure 8 shows the false positives for the standard deviation 12 operator. Figure 9 shows the false positive rate for the operator tuned to the current standard deviation of the noise. Figure 10 shows the false positive rate of the combined operator. Figure 11 shows the superposition of the 4 previous graphs.

[2]The operators are thresholded at 0.5 probability to make the decisions about where the boundaries are.

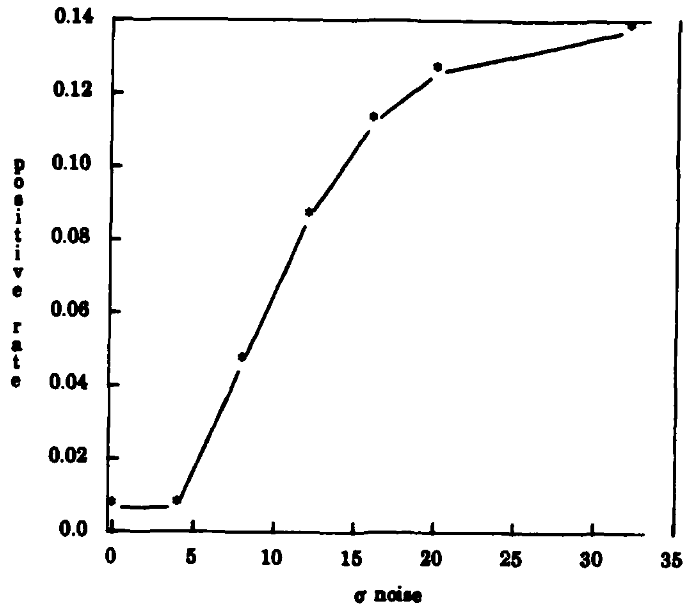


Figure 7: False positives vs noise σ for operator tuned to $\sigma = 4$ noise

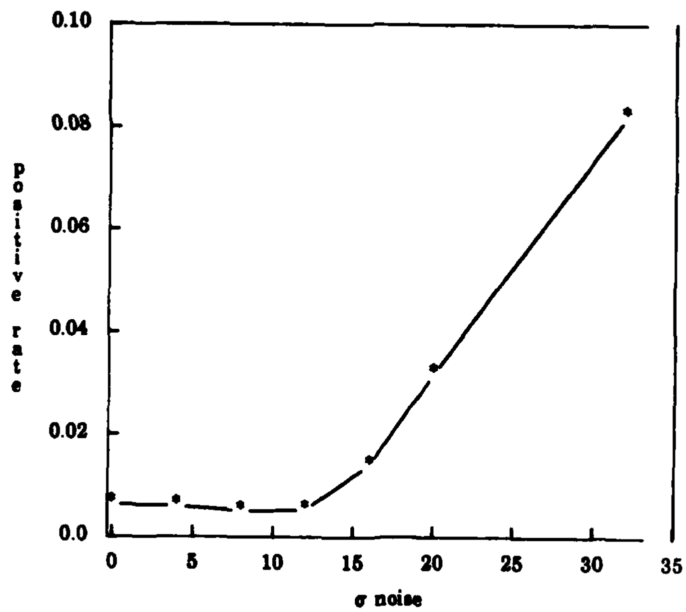


Figure 8: False positives vs noise σ for operator tuned to $\sigma = 12$ noise

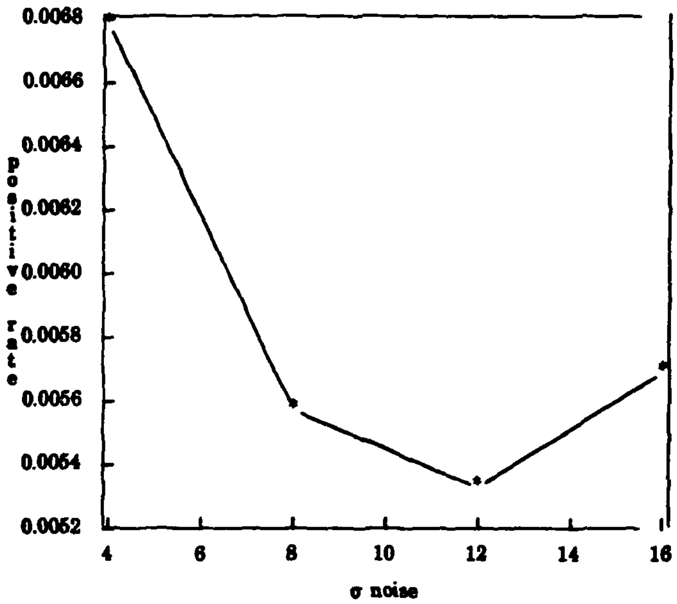


Figure 9: False positives vs noise σ for operator tuned to the noise

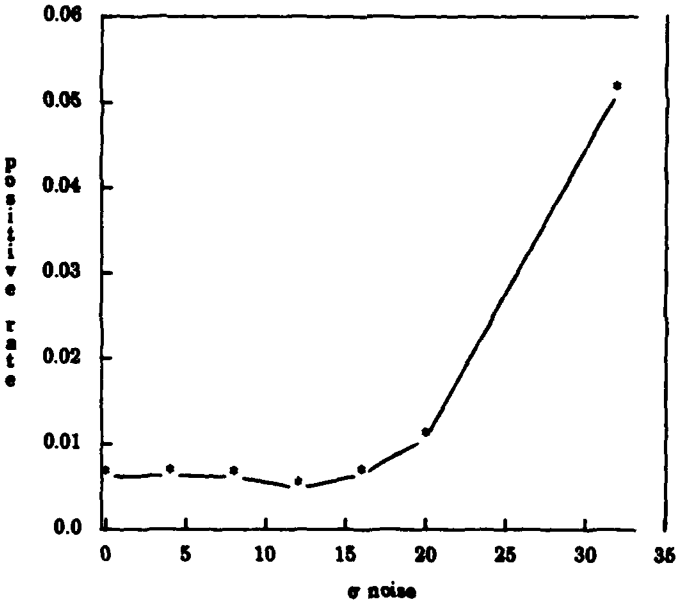
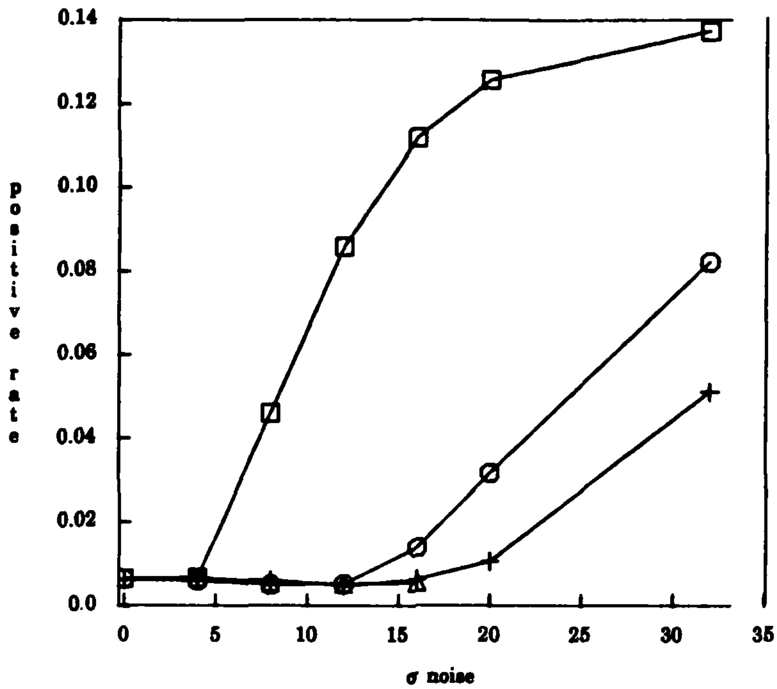


Figure 10: False positives vs noise σ for combined operator

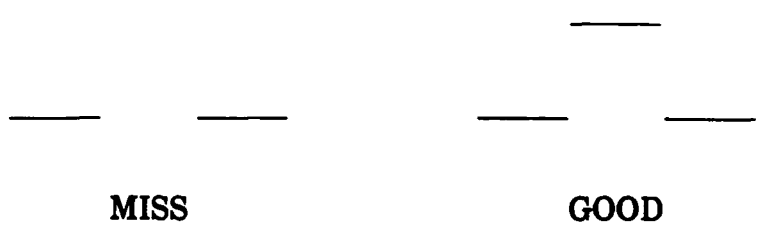


square: $\sigma=4$ operator circle: $\sigma=12$ operator
triangle: tuned operator cross: combined operator

Figure 11: False positives vs noise σ for all operators

Note that the combined operator has a false positive rate that is as least as good as that of the tuned operators.

I can also count false negatives. When I counted false negatives I ignored missed boundaries that had an boundary reported one pixel off normal to the boundary (because such an error is a matter of discretization rather than of a more fundamental sort). See figure 12 for an example of a 1 pixel off error.



MISS is recorded as a false negative
GOOD is recorded as a true positive

Figure 12: Example of one pixel off error

Figure 13 shows the false negative rate for the detector tuned to standard deviation 4 noise as the noise in the image increases. Figure 14 shows the false negatives for the standard deviation 12 operator. Figure 15 shows the false negative rate for the operator tuned to the current standard deviation of the noise. Figure 16 shows the false negative rate of the combined operator. Figure 17 shows the superposition of the 4 previous graphs.

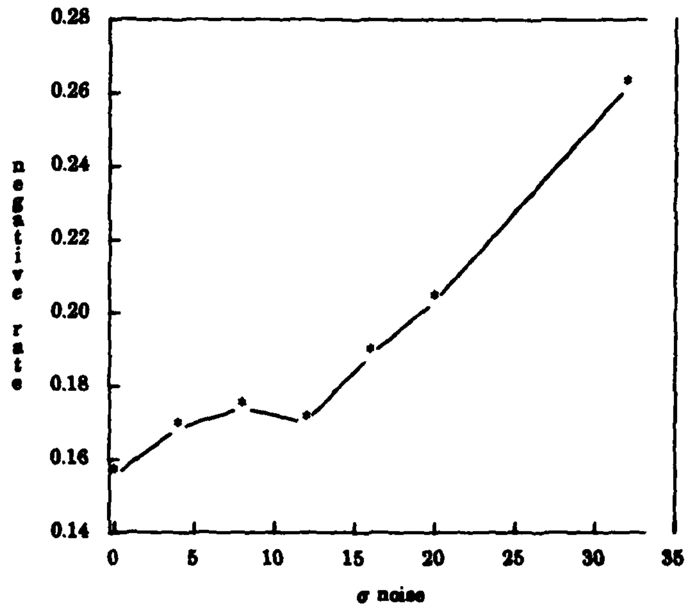


Figure 13: False negative rate for $\sigma = 4$ operator

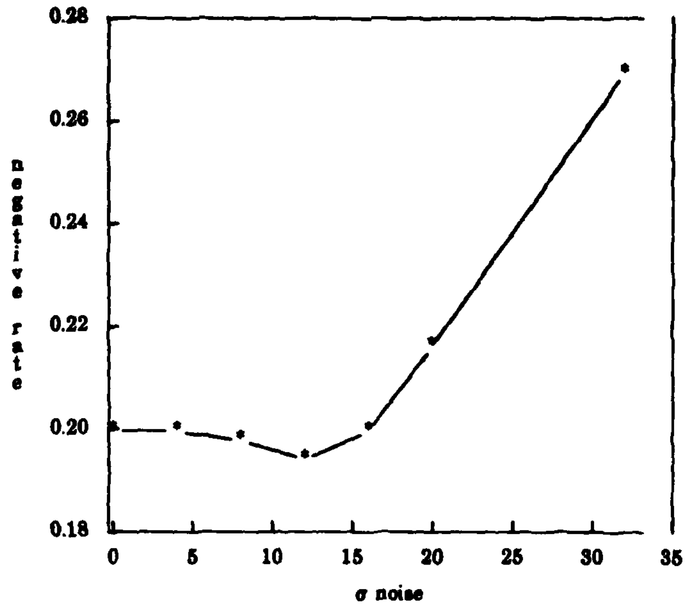


Figure 14: False negative rate for $\sigma = 12$ operator

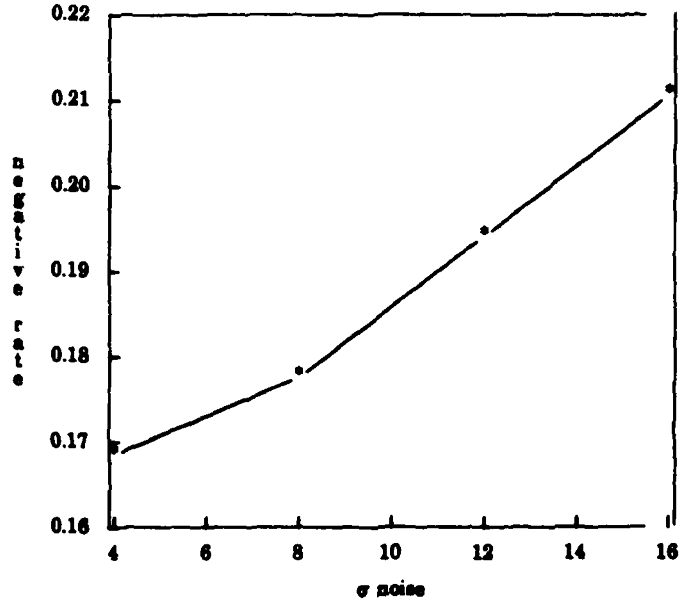


Figure 15: False negative rate for tuned operator

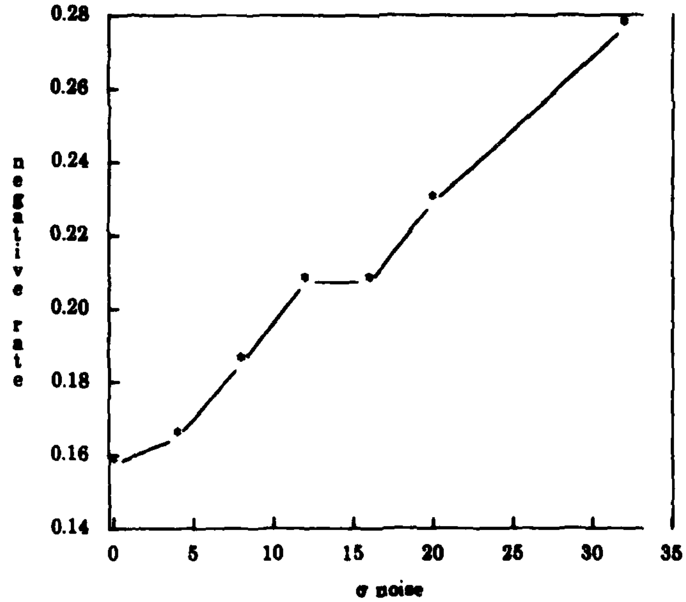
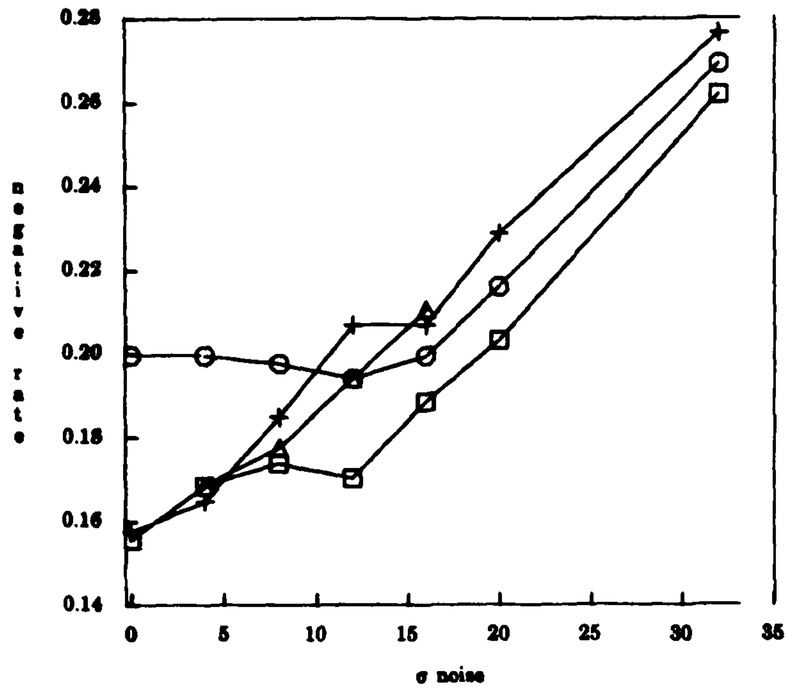


Figure 16: False negative rate for combined operator



square: $\sigma = 4$ operator circle: $\sigma = 12$ operator
triangle: tuned operator cross: combined operator

Figure 17: False negative rate for all operators

Here the combined operator is not always as good as the tuned operators. One must ask if this tendency of the combined operator to miss edges offsets its better performance for false positives. The next series of figures charts the total error rate for the same cases. Figure 18 shows the error rate for the detector tuned to standard deviation 4 noise as the noise in the image increases. Figure 19 shows the error rate for the standard deviation 12 operator. Figure 20 shows the error rate for the operator tuned to the current standard deviation of the noise. Figure 21 shows the error rate of the combined operator. Figure 22 shows the superposition of the 4 previous graphs.

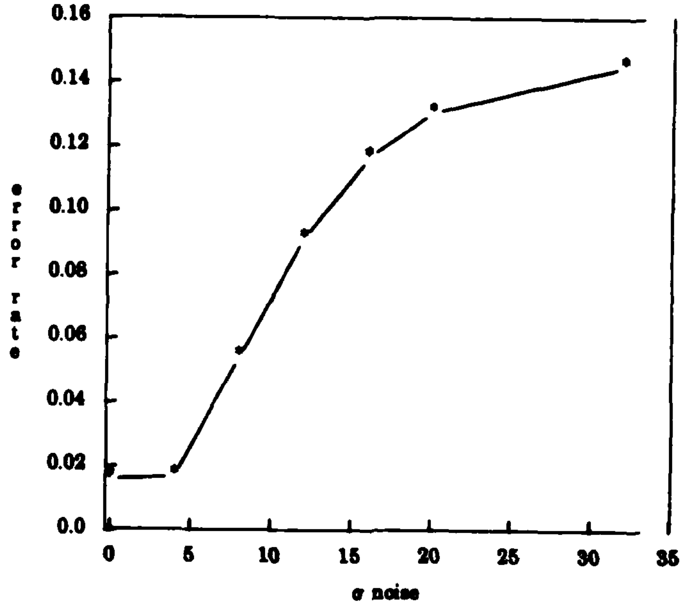


Figure 18: Total errors by the $\sigma = 4$ detector

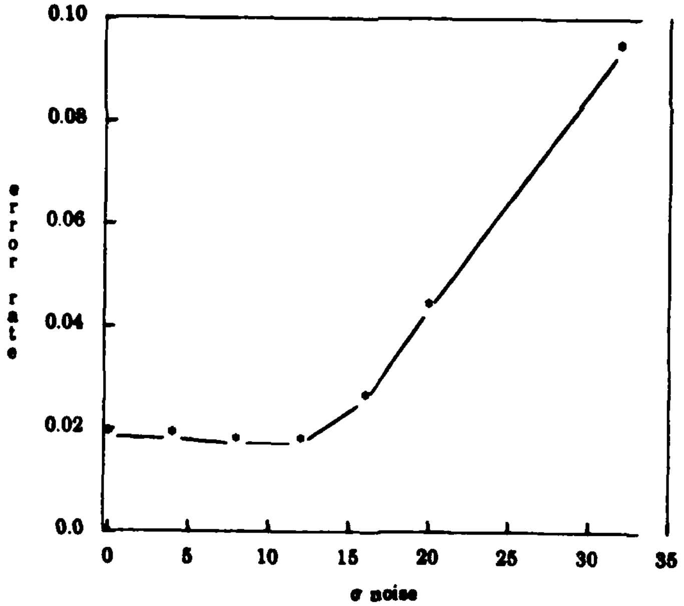


Figure 19: Total errors by the $\sigma = 12$ detector

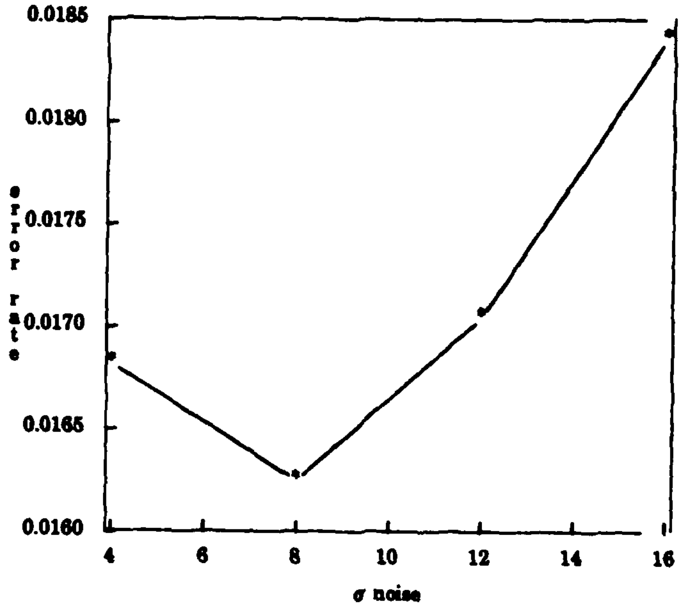


Figure 20: Total errors by the tuned detector

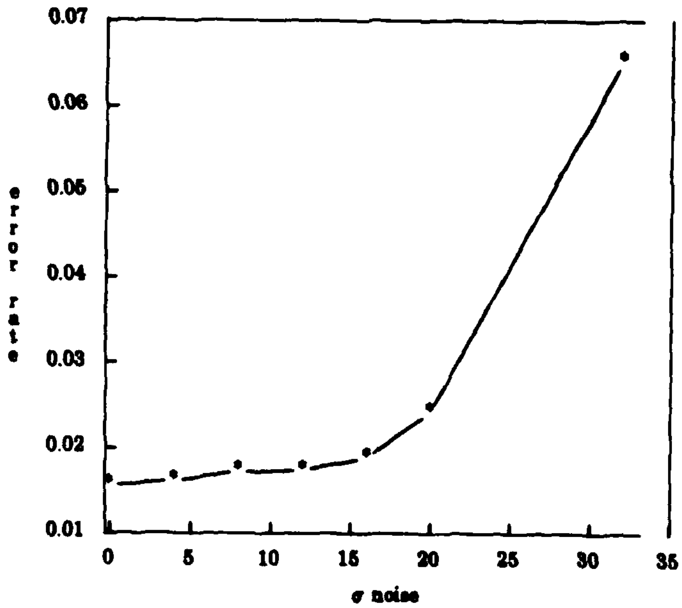
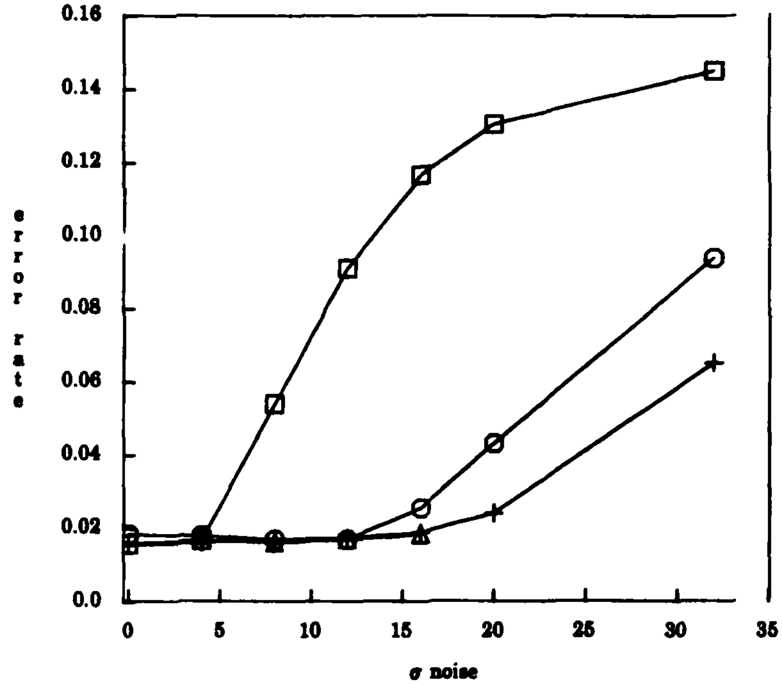


Figure 21: Total errors by the combined detector



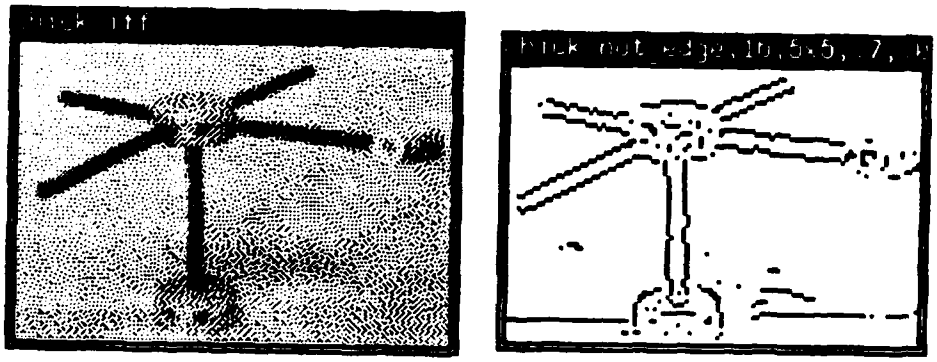
square: $\sigma = 4$ operator circle: $\sigma = 12$ operator
triangle: tuned operator cross: combined operator

Figure 22: Total errors by the all detectors

Thus the superiority of the combined operator for false positives dominates the false negative performance and the combined operator minimizes the number of errors in total. These results are evidence that my combination rule is robust.

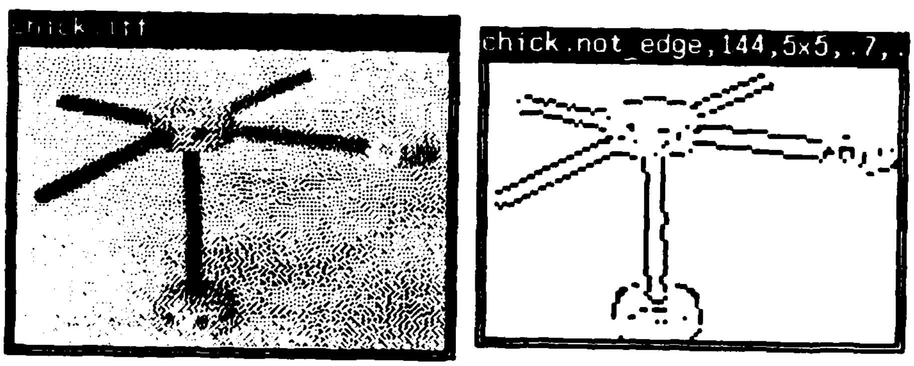
5.2. Real Images

I have also tested these theories using two images taken by cameras. One of these images is a tinker toy image taken in our lab. The other is an aerial image of the vicinity of Lake Ontario. Figure 23 shows the result of the operator tuned to standard deviation 4 noise applied to the tinker toy image and thresholded at 0.5 probability. Figure 24 shows the result of the operator tuned to standard deviation 12 noise applied to the tinker toy image. Figure 25 shows the effect of combining operators tuned to standard deviation 4, 8, 12 and 16 with equal probability.



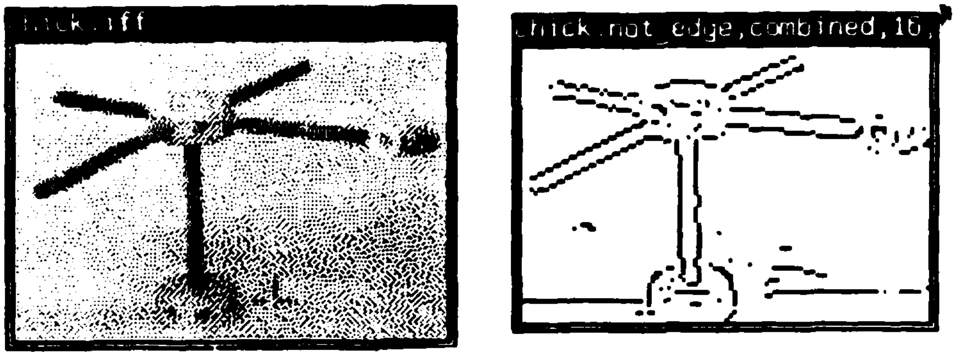
a: Tinkertoy Image   b: Output of $\sigma=4$ detector

Figure 23: $\sigma=4$ detector applied to tinkertoy image



a: Tinkertoy Image   b: Output of $\sigma=12$ detector

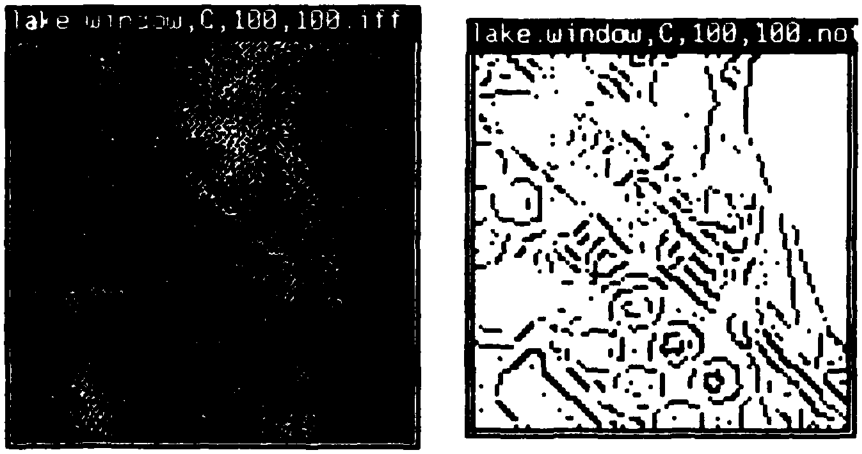
Figure 24: $\sigma=12$ detector applied to tinkertoy image



a: Tinkertoy Image   b: Output of combined detector
Figure 25: Combined detector applied to tinkertoy image

Here, the result of the combined operator seems to be a cleaned up version of the standard deviation 4 operator. Most of the features that are represented in the output of the combined operator are however real features of the scene. The line running horizontally across the image that the standard deviation 4 operator and the combined operator found is the place where the table meets the curtain behind the tinkertoy. The standard deviation 4 operator was certain of its interpretation and the other operators were uncertain at that point so its interpretation was used by the combination.

The results from the aerial image are also instructive. Figure 26 shows the result of the operator tuned to standard deviation 4 noise applied to the aerial image and thresholded at 0.5 probability. Figure 27 shows the result of the operator tuned to standard deviation 12 noise applied to the aerial image. Figure 28 shows the effect of combining operators tuned to standard deviation 4, 8, 12 and 16 with equal probability.



a: Aerial Image b: Output of $\sigma=4$ detector
Figure 26: $\sigma=4$ detector applied to aerial image

a: Aerial Image b: Output of $\sigma=12$ detector
Figure 27: $\sigma=12$ detector applied to aerial image

a: Aerial Image b: Output of combined detector
Figure 28: Combined detector applied to aerial image

The results from the combined operator are again a cleaned up version of the results from the standard deviation 4 operator. I believe this behavior occurs again because the features being found by the standard deviation 4 operator are in the scene. However I do not have the ground truth for the aerial image as I do for the tinkertoy image.

5.3. Future Experiments

Soon, I will apply my evidence combination rules to operators that make different assumptions about the expected image intensity histogram. The operator used so far in my experiments expects a uniform histogram between 0 and 254. Currently, a likelihood generator has been built that assumes a triangular distribution with the probability of an object having intensity less than 128 being one fourth the probability of an object having intensity greater than or equal to 128. It is not clear that the probabilities calculated based on this assumption will be significantly different from those based on the uniform histogram assumption. If there is no difference in the output of two operators the effect of combination is invisible.

Larger operators will soon be available. The likelihoods generated based on these larger operators would be finely tuned. The same evidence combination can be applied to these operators.

Likelihoods are used by Markov random field algorithms to determine posterior probabilities [Marroquin85b] [Chou87]. Likelihoods resulting from my combination rules can be used by Markov random field algorithms.

6. Previous Work

Much of the work on evidence and evidence combination in vision has been on high level vision. An important Bayesian approach (and a motivation for my work) was by Feldman and Yakimovsky [Feldman74]. In this work Feldman and Yakimovsky were studying region merging based on high level constraints. They first tried to find a probability distribution over the labels of a region using characteristics such as mean color or texture. They then tried to improve these distributions using labelings for the neighbors. Then they made merge decisions based on whether

it was sufficiently probable that two adjacent regions were the same.

Work with a similar flavor has been done by Hanson and Riseman. In [Hanson80] Bayesian theories are applied to edge relaxation. This work had serious problems with its models and the fact that the initial probabilities input were edge strengths normalised never to exceed 1. Of course such edge strengths have little relationship to probabilities (a good edge detector tries to be monotonic in its output with probability but that is about as far as it gets). In [Wesley82a] and [Wesley82b] Dempster-Shafer evidence theory is used to model and understand high level problems in vision especially region labeling. In [Wesley82b] there is some informed criticism of Bayesian approaches. In [Reynolds85] They study how one converts low level feature values into input for a Dempster-Shafer evidence system.

In [Levitt85] Tod Levitt takes an approach to managing a hierarchical hypothesis space that is baysian with some ad hoc assumptions. For the problem worked on here the paper would take weighted sums of probabilities. He does not have any way of taking an operators self confidence into account in the evidence combination. Since he was not approaching this problem in his paper I can not fault it in this respect.

There has been much use of likelihoods in recent vision work. In particular work based on Markov random fields [Geman84] [Marroquin85a] [Marroquin85b] use likelihoods. A Markov random field is a prior probability distribution for some feature of an image and the likelihoods are used to compute the marginal posterior probabilities that are used to update the field. Haralick has mentioned that his facet model [Haralick84] [Haralick86b] can be easily used to build edge detectors that return likelihoods [Haralick86a]. I also have built boundary detectors that return likelihoods and the results of using them is documented in [Sher87]. Paul Chou is using the likelihoods I produce with Markov random fields for edge relaxation [Chou87]. He is also studying the use of likelihoods for information fusion. Currently, he is concentrating on information fusion from different sources of information.

7. Conclusion

I have presented a Bayesian technique for information fusion. I show how to fuse information from detectors with different models. I presented results from applying these techniques to artificial and real images.

These techniques take several operators that are tuned to work well when the scene has certain particular properties and get an algorithm that works almost as well as the best of the operators being combined. Since most algorithms available for machine vision are erratic when their assumptions are violated this work can be used to improve the robustness of many algorithms.

References

[Aloimonos85]
J. Aloimonos and P. Chou, Detection of Surface Orientation and Motion from Texture: 1. The Case of Planes, 161, Computer Science Department, University of Rochester, January 1985.

[Ballard82]
D. H. Ballard and C. M. Brown, in *Computer Vision*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1982, 125.

[Chou87] P. Chou, Multi-Modal Segmentation using Markov Random Fields, *Submitted to IJCAI*, January 1987.

[Feldman74]
J. A. Feldman and Y. Yakimovsky, Decision Theory and Artificial Intelligence: I. A Semantics-Based region Analyzer, *Artificial Intelligence 5*(1974), 349-371, North-Holland Publishing Company.

[Frieden85]
B. R. Frieden, Estimating Occurrence Laws with Maximum Probability, and the Transition to Entropic Estimators, in *Maximum-Entropy and Bayesian Methods in Inverse Problems*, C. R. Smith and W. T. G. Jr. (editor), D. Reidel Publishing Company, Lancaster, 1985.

[Geman84]
S. Geman and D. Geman, Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, *PAMI 6*,6 (Novermber 1984), 721-741, IEEE.

[Good50] I. J. Good, *Probability and the Weighing of Evidence*, Hafner Publishing Company., London, New York, 1950

[Good83] I. J. Good, Subjective Probability as the Measure of a Non-measurable Set, in *Good Thinking: The Foundations of Probability and its Applications*, Minneapolis (editor), University of Minnesota Press, Minneapolis, 1983, 73-82.

[Hanson80]
A. R. Hanson, E. M. Riseman and F. C. Glazer, Edge Relaxation and Boundary Continuity, 80-11, University of Massachusetts at Amherst, Computer and Information Science, May 1980.

[Haralick84]
R. M. Haralick, Digital Step Edges from Zero Crossing of Second Directional Derivatives, *PAMI 6*,1 (January 1984), 58-68, IEEE.

[Haralick86a]
R. Haralick, Personal Communication, June 1986.

[Haralick86b]
R. M. Haralick, The Facet Approach to Gradient Edge Detection, *Tutorial 1 Facet Model Image Processing (CVPR)*, May 1986.

[Horn70] B. K. P. Horn, *Shape from Shading: A Method for FInding the SHape of a Smooth Opaque Object from One View*, Massachusetss Institute of Technology Department of ELectrical Engineering., August 1970

[Ikeuchi80]
K. Ikeuchi, Shape form Regular Patterns (an Example of Constraint Propagation in Vision), 567, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, March 1980.

[Johnson85]
R. W. Johnson and J. E. Shore, Introduction to Minimum-Cross-Entropy Spectral Analysis of Multiple Signals, in *Maximum-Entropy and Bayesian Methods in Inverse Problems*, C. R. Smith and W. T. G. Jr. (editor), D. Reidel Publishing Company, Lancaster, 1985.

[Levitt85] T. S. Levitt, Probabilistic Conflict Resolution in Hierarchical Hypothesis Spaces, *Proceedings: Uncertainty and Probability in Artificial Intelligence*, August 14-16, 1985, 265 272.

[Marroquin85a]
J. Marroquin, S. Mitter and T. Poggio, Probabilistic Solution of Ill-Posed Problems in Computational Vision, *Proceedings: Image Understanding* Workshop, December 1985, 293-309. Sponsored by: Information Processing Techniques Office Defence Advanced Research Projects Agency.

[Marroquin85b]
J. L. Marroquin, Probabilistic Solution of Inverse Problems, Tech. Rep. 860, MIT Artificial Intelligence Laboratory, September 1985.

[Ohlander79]
R. Ohlander, K. Price and D. R. Reddy, Picture Segmentation using a Recursive Region Splitting Method, *CGIP 8*,3 (1979).

[Reynolds85]
G. Reynolds, D. Strahman and N. Lehrer, Converting Feature Values to Evidence, *PROCEEDINGS: IMAGE UNDERSTANDING WORKSHOP*, December 1985, 331-339. Sponsored by: Information Processing Techniques Office, Defence Advanced Research Projects Agency.

[Sher86] D. Sher, Optimal Likelihood Detectors for Boundary Detection Under Gaussian Additive Noise, *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, Florida, June 1986.

[Sher87] D. B. Sher, Advanced Likelihood Generators for Boundary Detection, TR197, University of Rochester Computer Science Department, London, England, January 1987. Submitted in shorter form to International Conference on Computer Vision.

[Wesley82a]

L. P. Wesley and A. R. Hanson, The Use of an Evidential-Based Model for Representing Knowledge and Reasoning about Images in the Visions System, *PAMI 4*,5 (Sept 1982), 14-25, IEEE.

[Wesley82b]

L. P. Wesley and A. R. Hanson, The use of an Evidential-Based Model for Representing Knowledge and Reasoning about Images in the VISIONS System, *Proceedings of the Workshop on Computer VIsion: Representation and Control*, August 1982, 14-25.

END

5_87

DTIC